SISTEM KEAMANAN PESAN DENGAN ALGORITMA RIVEST CODE 6 (RC-6) MENGGUNAKAN JAVA PADA SMARTPHONE BERBASIS ANDROID

Dr. H. Riza M Yunus, Harun Sujadi, Karnia
Teknik Informatika, Fakultas Teknik Universitas Majalengka
email: riza_majalengka@yahoo.co.id harunsujadi@gmail.com nia_opick@yahoo.com

ABSTRACT

Rapid expansion of telecommunication technology has brought a very big benefit for us. In an aspect of information system security is one aspect that is very important. One mechanism that is done to secure this is by cryptographic techniques with methods of encryption and decryption using the algorithm Rivest Code 6 (RC-6) using Java in Android-Based Smartphone.With the existence of telecommunication technology, many constraints such as distance, location, or time can be overcome. One of the technological results in telecommunication technology is Short Message Service or usually known as SMS. By using an SMS, the subscribers can do some exchange of text messages over each other. In this final assignment, developed an aplication on a cell phone to modify a text into ciphertext so that information from the message is not known by others. For sending an SMS, system encrypts plaintext into ciphertext using a key that is inputed by a sender and then send it to destination number. For receiving an SMS, system decrypt ciphertext into plaintext using a key that is inputed by receiver and then displaying plaintext to receiver. This application can be used by someone who wants to send a secret information to other through an SMS without fear of information from those messages will be known by others. RC6 is one of the most sophisticated cryptographic algorithms and can still be said to be unresolved. The application uses the algorithm Rivest Code 6 (RC-6) with the Java programming language as data encryption method that can be applied to Android-based smartphones.

Keywords: *Message Security Systems, Algorithm RC-6, Android.*

I. PENDAHULUAN

A. Latar Belakang Masalah

Beberapa tahun terakhir ini terjadi perkembangan yang pesat pada teknologi, salah satunya adalah telepon selular (ponsel). Mulai dari ponsel yang hanya bisa digunakan untuk bicara dan sms hingga "ponsel cerdas" (*smartphone*) yang memiliki berbagai fungsi seperti multimedia, multiplayer games, transfer data, video streaming dan lain – lain.

Berbagai perangkat lunak untuk mengembangkan aplikasi ponselpun bermunculan, diantaranya yang cukup luas adalah android. Salah satu fasilitas yang disediakan ponsel adalah untuk melakukan pengiriman data berupa pesan singkat melalui *short meesage service* (SMS). Namun dengan fasilitas SMS yang ada, timbul pertanyaan mengenai keamanan informasi jika seseorang ingin mengirimkan suatu informasi rahasia melalui fasilitas SMS. Di Indonesia sekarang sudah ada yang namanya Flash Massage yaitu pesan datang dan pergi cepat kilat, yaitu setelah di balas dan menekan tombol keluar secara otomatis sms-nya akan hilang dari inbox tanpa jejak.

Perlindungan data konsumen adalah perangkat hukum yang diciptakan untuk melindungi dan terpenuhinya hak konsumen yang dijelaskan dalam Undang-undang Perlindungan Konsumen Nomor 8 Tahun 1999 tentang Perlindungan Konsumen Republik Indonesia menjelaskan bahwa hak konsumen diantaranya adalah hak atas kenyamanan, keamanan, dan keselamatan dalam

mengkonsumsi barang dan atau jasa. dan dapat didefinisikan sebagai " segala upaya yang menjamin adanya kepastian hukum untuk memberikan perlindungan kepada konsumen".

Diluar negeri pemanfaatan SMS untuk mengirimi pesan rahasia telah lebih dulu dikembangkan. Misalnya di inggris sebuah perusahaan operator telepon selular, staellium UK, mengeluarkan layanan bernama "*stealth text"* yang dapat digunakan untuk mengirim pesan dengan aman, yaitu dengan cara menghapus pesan secara otomatis segera setelah 40 detik pesan dibaca atau yang dikenal dengan nama *self-destruct text message*.

Ada juga pengamanan sms dengan menggunakan kriptografi sms yang memanfaatkan kunci untuk mendekripsikan sms yang telah di enkripsi.

B. Rumusan Masalah

Berdasarkan uraian pada latar belakang di atas, maka didapatkan rumusan masalah, yaitu :

1. Bagaimana merancang sistem keamanan pesan menggunakan algoritma rivest code 6 (RC-6) menggunakan java?
2. Bagaimana merancang sistem keamanan pesan agar dapat diterapkan pada smartphone berbasis android?
3. Bagaimana sistem keamanan pesan tersebut dirancang sehingga mudah untuk digunakan?

C. Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut :

1. Sistem keamanan pesan ini membuat pesan informasi untuk dikirim dan juga menampilkan pesan informasi yang diterima.
2. Sistem keamanan pesan ini menggunakan algoritma rivest code 6 (RC-6) sebagai metode enkripsi datanya.
3. Sistem keamanan pesan ini dapat diterapkan hanya pada smartphone yang berbasis android.

II. LANDASAN TEORI

Secara umum, data dapat didefinisikan sebagai hasil dari pengolahan fakta yang jauh lebih berharga ditambah jauh lebih bermakna untuk penerima informasi yang menjelaskan kesempatan nyata yang digunakan untuk pengambilan penentuan keputusan. Informasi dapat berupa fakta yang telah diberi label atau mungkin diproses atau dilihat dan berkaitan dengan memanfaatkan dalam tindakan yang melibatkan penentuan pengabilan keputusan.

Suatu sistem dapat terdiri dari beberapa subsistem atau sistem-sistem bagian. Komponen-komponen atau subsitem dalam suatu sistem tidak dapat berdiri lepas sendiri-sendiri. Komponen-komponen dan subsistem saling berinteraksi dan saling berhubungan membentuk satu kesatuan sehingga tujuan atau sasaran dapat tercapai.

A. Java

Java adalah bahasa pemograman yang dapat dijalankan diberbagai computer termasuk telepon genggam. Dikembangkan oleh Sun Microsystems dan dirilis tahun 1995. Java berbeda dengan JavaScript. JavaScrip adalah bahasa scripting yang digunakan oleh web browser.

Java adalah sebuah bahasa pemrograman komputer berbasiskan kepada Object Oriented Programming. Java didesain sedemikian rupa sehingga ukurannya kecil, sederhana, dan portable (dapat dipindah-pindahkan di antara bermacam platform dan sistem operasi). Program yang dihasilkan dengan bahasa java dapat berupa applet (aplikasi kecil yang jalan di atas web browser) maupun berupa aplikasi mandiri yang dijalankan dengan program java Interpreter.

Bahasa pemograman java pertama lahir dari The Green Project, yang berjalan selama 18 bulan, dari awal tahun 1991 hingga musim panas 1992. Proyek tersebut belum menggunakan versi yang dinamakan Oak. Proyek ini dimotori oleh Patrik Naughton, Mike Sheridan, James Gosling dan Bill Joy, beserta Sembilan pemogram lainnya dari Sun Microsystems. Salah satu hasil proyek ini adalah mascot Duke yang dibuat oleh Joe Palang.

Mereka menjadikan parambah (browser) Mosaic sebagai landasan awal untuk membuat perambah Java pertama yang dinamakan Web Runner, terinspirasi dalam film 1980-an, Blade Runner. Pada perkembangan rilis pertama, Web Runner berganti nama menjadi Hot Java.

Pada sekitar bulan Maret 1995, untuk pertama kali kode sumber Java versi 1.0a2 dibuka. Kesuksesaan mereka diikuti dengan untuk pemberitaan pertama kali pada surat kabar San Joe Mercury News pada tanggal 23 Mei 1995. Nama Oak

ini tidak di pakai untuk versi release Java karena sebuah perangkat lunak lain sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi " Java ". Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan Gosling. Konon kopi ini berasal dari Pulau Jawa. Jadi nama bahasa pemograman Java tidak lain berasal dari kata Java (bahasa Inggris untuk Java adalah Jawa).

B. Eclipse

Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak IBM Visual Age for Java. Produk ini diluncur oleh IBM pada tanggal 5 November 2001, yang menginvestasikan sebanyak U$$ 40juta, untuk pengembangannya.

Konsep Eclipse adalah IDE yang terbuka *(open)*, mudah diperluas (*extensible*) untuk apa saja dan tidak untuk sesuatu yang spesifik, jadi Eclipse tidak saja untuk mengembangkan program java, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal *plug-in* yang dibutuhkan.

Selain pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, plug-in tersedia untuk membuat diagram UML. Dengan menggunakan PDE setiap orang bias membuat plug-in sesuai dengan keinginan

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform-independent*).

C. Android

Android adalah sebuah sistem operasi untuk perangkat lunak *mobile* berbasis linux yang mencakup sistem operasi, middleware dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel / smartphone. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, Htc, Intel, Motorola, Qualcomm, T-*Mobile*, dan Nvidia.

Pada saat perilisan perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan *open source* pada perangkat *mobile*. Di lain pihak, Google merilis kode - kode android dibawah lisensi *Apache*, sebuah lisensi perangkat lunak dan open *platform* perangkat seluler.

Didunia ini terdapat dua jenis distributor sistem operasi android. Pertama yang mendapat dukungan penuh dari Google atau Google Mail *Service*s (GMS) dan kedua adalah yang benar - benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD).

Tidak hanya menjadi sistem operasi di smartphone, saat ini android menjadi pesaing utama dari Apple pada sistem operasi Table PC. Pesatnya pertumbuhan Android selain faktor yang disebutkan diatas adalah karena android itu sendiri adalah *platform* sangat lengkap baik itu sistem operasinya, aplikasi dan Tool Developmen, Market aplikasi android serta dukungan yang sangat tinggi dari komunitas *Open source* didunia, sheingga android terus berkembang pesat dari segi teknologi maupun dari segi jumlah device yang ada didunia.

D. Algoritma RC-6

RC6 merupakan algoritma cipher blok baru yang didaftarkan ke NIST yang diajukan oleh RSA Security Laboratories. Algoritma ini dirancang oleh Ronald L Rivest, M.J.B. Robshaw, R. Sidney dan Y.L. Yin untuk mengikuti kontes Advanced Ecryption Standard (AES) dan berhasil menjadi salah satu dari lima (5) finalisnya. Design dari berawal dari keinginan untuk meningkatkan performansi dan tingkat keamanan dari RC5 untuk dapat memenuhi standar dari kontes tersebut.

RC6 memiliki struktur yang sederhana. RC6 terdiri dari dua jaringan Feistel dimana datanya dicampur dengan rotasi yang bergantung pada isi data tersebut. Dalam sekali putaran RC6, ada beberapa operasi yang terjadi, antara lain : dua (2) aplikasi dari fungsi persamaan $f(x) = x(2x + 1) \bmod 232$, dua (2) rotasi 32-bit yang tidak berubah, dua (2) rotasi 32-bit yang bergantung pada data, dua (2) eksklusif OR dan dua (2) fungsi modulo 232 tambahan. Algoritma cipher ini biasanya memakai 20 putaran.

RC6, bila dibandingkan dengan RC5, menggunakan 4 (empat) working registers, dan menyertakan operasi perkalian integer sebagai operasi primitif tambahan. Operasi perkalian

meningkatkan penyebaran untuk tiap putarannya sehingga meningkatkan faktor keamanan, mengurangi putaran, dan meningkatkan performa hasil.

Tingkat keamanan pada algoritma ini terletak pada kekuatan rotasi yang berdasarkan data, penggunaan eksklusif OR yang bergantian, fungsi modulo dan fungsi persamaan yang menggunakan rotasi yeng tetap. Dengan menghilangkan salah satu atau beberapa aspek di atas, maka cipher yang dihasilkan akan menjadi lebih lemah terhadap beberapa serangan yang sudah diketemukan sebelumnya. Beberapa jenis serangan modern terhadap algoritma ini hanya dapat dilakukan secara teori tanpa praktek serangan sesungguhnya.

E. Deskripsi

Algoritma RC6 adalah suatu algoritma kriptografi *block cipher* yang dirancang oleh Ronald L. Rivest, Matt J.B. Robshaw, Ray Sidney, dan Yuqin Lisa Yin dari RSA Laboratories. Algoritma ini pada mulanya dirancang untuk menjadi AES (*Advance Encryption Standard*). Algoritma RC6 ini berhasil menjadi finalis dan menjadi kandidat kuat untuk menjadi AES walaupun pada akhirnya algoritma ini tidak terpilih menjadi AES melainkan algoritma *rinjdael*. Versi 1.1 dari RC6 mulai dipublikasikan pada tahun 1998. Dasar desain dari algoritma RC6 ini didasarkan pada pendahulunya yaitu algoritma RC5.

Desain algoritma RC5 mengutamakan kesederhanaan agar mudah untuk diimplementasikan, selain itu juga kecepatan dan penggunaan memori yang rendah menjadi faktor utama perancangan algoritma RC5. Algoritma RC5 dirancang agar dapat beradaptasi dengan prosesor yang beragam dan juga didesain dengan struktur yang iteratif dengan jumlah iterasi yang dapat beragam, sehingga algoritma RC5 memiliki parameter agar dapat bekerja dengan jumlah iterasi dan blok yang beragam. Algoritma RC5 bekerja dengan dua buah register A dan B sebesar panjang blok dibagi dua, proses enkripsi dari algoritma RC5 dengan S adalah *array* yang berisi kunci internal dan r adalah jumlah iterasi adalah sebagai berikut:

$A \leftarrow A + S[0]$
$B \leftarrow B + S[1]$
for i ← 1 to r do
    $A \leftarrow ((A \oplus B) <<< B) + S[2*i]$
    $B \leftarrow ((B \oplus A) <<< A) + S[2*i+1]$
endfor

Proses dekripsi algoritma RC5 adalah sebagai berikut:

for i ← r downto r do
    $B \leftarrow ((B - S[2*i+1]) >>> A) \oplus A$
    $A \leftarrow ((A - S[2*i]) >>> B) \oplus B$
endfor
$B \leftarrow B - S[1]$
$A \leftarrow A - S[0]$

Seperti halnya algoritma RC5, algoritma RC6 merupakan algoritma dengan parameter penuh, algoritma RC6 dispesifikasikan dengan notasi RC6-*w/r/b.* Dimana *w* adalah ukuran dari *word* dalam bit, karena pada RC6 menggunakan 4 buah register maka *word* adalah ukuran blok dibagi 4. *r* adalah jumlah iterasi, dimana *r* tidak boleh negatif. Dan *b* adalah panjang kunci dalam *bytes*. Dalam rancangan untuk menjadi kandidat AES algoritma RC6 yang digunakan menggunakan ukuran *w* sebesar 32 bit dan jumlah iterasi *r* sebesar 20 kali putaran.

Cara kerja dari algoritma RC6 adalah menggunakan 4 buah register dan menggunakan prinsip *Iterated Block Cipher* yang mengunakan iterasi, dalam algoritma ini tidak digunakan S-box.

F. Pembentukan Kunci Internal

Untuk membangkitkan urutan kunci internal yang akan digunakan selama proses enkripsi, algoritma RC6 melakukan proses pembangunan kunci yang identik dengan algoritma RC5, yang membedakan hanyalah pada algoritma RC6, jumlah *word* yang diambil dari kunci yang dimasukan oleh pengguna ketika melakukan enkripsi ataupun dekripsi lebih banyak. Tujuan dari proses pembangunan kunci tersebut adalah untuk membangun suatu *array* S yang berukuran 2*r*+4 dari kunci masukan pengguna sepanjang *b bytes* (0 _ *b* _ 255), array tersebut akan digunakan baik dalam proses enkripsi maupun dekripsi.

Proses untuk membangun kunci-kunci internal menggunakan dua buah konstanta yang disebut dengan "*magic constant*". Dua buah *magic constant* *Pw* dan *Qw* tersebut didefinisikan sebagai berikut:

Pw = *Odd*((*e*-2)2*w*)……..*(2.1)*

Qw = *Odd*((-1)2*w*)……..*(2.2)*

Dimana :

e = 2.7182818284859…(basis dari logaritma natural)

= 1.618022988749…(golden ratio)

Odd (x) adalah integer ganjil terdekat dari x, jika x genap maka diambil integer ganjil setelah x.

Berikut adalah daftar *magic constant* pada beberapa panjang blok dalam heksadesimal:

P16 = b7e1
Q16 = 9e37
P32 = b7e15163
Q32 = 9e3779b9
P64 = b7e151628aed2a6b
Q64 = 9e3779b97f4a7c15

Dengan menggunakan dua buah *magic constant* tersebut, pembangunan kunci terdiri dari tiga tahap :

1. Konversi kunci rahasia dari *bytes* ke *words*

Langkah pertama adalah menyalin kunci rahasia K[0..b-1] kedalam sebuah *array* L[0..c-1], dimana c = pembulatan keatas(*b*/u) dan u = *w*/8, penyalinan tersebut dilakukan secara *little endian.* Untuk semua posisi *byte* pada L yang kosong diberi nilai nol. Untuk kasus dimana *b* = 0, maka *c* = 1 dan L[0] = 0. Langkah ini dapat dilakukan dengan cara berikut :

```
if c=0 then
    c ← 1
endif
for i ←b-1 downto 0 do
    L[i/u] ← (L[i/u] <<<8) + K[i]
endfor
```

2. Inisialisasi *array* S

Langkah kedua adalah melakukan inisialisasi array S agar memiliki pola *pseudo*-andom bit tertentu menggunakan progresi aritmatika modulo 2*w* yang ditentukan dengan *Pw* dan *Qw*. Berikut langkah kedua dalam pseudo code :

```
S[0] ← Pw
for i ← 0 to 2r+3 do
    S[i] ← S [i-1] + Qw
endfor
```

3. Mencampurkan L dan S

Langkah terakhir adalah mencampurkan kunci rahasia dari pengguna yang sudah tersimpan dalam L dengan S sebanyak 3 kali iterasi. Berikut adalah langkah pencampuran tersebut.

```
I ← 0
J ← 0
A ← 0
B ← 0
V ← 3*max (c, 2r+4)
for index ← 1 to v do
    S[i] ← (S[i]+A+B) <<< 3
    A ← S[i]
    L[j] ← (L[j]+A+B) <<<
(A+B)
    B ← L[j]
    i ← (i+1) mod (2r+4)
    j ← (j+1) mod c
endfor
```

Pembentukan kunci yang dilakukan, mengubah kunci dari *user* yang panjangnya beragam (0-255) menjadi suatu rangkaian kunci dengan sepanjang *word* sebanyak 2*r+3* buah. Hal ini menjadikan RC6 dapat bekerja dengan kunci masukkan pengguna yang beragam.

Kunci yang dihasilkan oleh proses pembentukian kunci ini memiliki sifat satu arah, sehingga proses pembentukan kunci ini dapat digunakan sebagai fungsi *hash* satu arah. Dengan sifat satu arah tersebut, maka kunci internal akan sangat berbeda dengan kunci yang dimasukkan oleh pengguna, hal ini akan membuat hubungan statistik antara kunci yang dimasukan oleh pengguna dengan plainteks dan cipherteks menjadi lebih rumit karena dalam melakukan enkripsi, kunci yang dipakai adalah kunci internal.

Pada pembentukan kunci internal digunakan iterasi yang cukup banyak baik pada tahap satu, dimana untuk melakukan ekspansi kunci dibutuhkan iterasi, dan pada tahap dua, dimana dibutuhkan iterasi untuk melakukan inisialisai array serta pada tahap terakhir yang dibutuhkan untuk menggabungkan dua buah array, yang bahkan dilakukan selama tiga kali. Iterasi-iterasi ini membutuhkan waktu yang cukup besar untuk dilakukan.

G. Kriptografi

Mekanisme keamanan jaringan pada implementasi menggunankan teknikteknik penyandian yaitu kriptografi.

Kriptografi mempunyai tujuan yaitu mengamankan isi data atau menjaga kerahasian informasi dari orang yang tidak berhak untuk mengetahui isi data tersebut. Agar isi data aman maka diperlukan teknik atau algoritma untuk mengamankannya seperti proses enkripsi dan Dekripsi , untuk dapat melakukan proses tersebut maka pengirim dan penerima harus mengetahui algoritma yang digunakan dan memiliki kunci yang sesuai.

Tingkat keamanan dari data sandi terhadap upaya proses deskirpsi secara paksa oleh orang yang tidak berhak ditentukan oleh kekuatan algoritma yang digunakan dan kerahasian kunci. Kekuatan algoritma yang digunakan untuk proses enkripsi dan deskipsi berhubungan erat dengan penggunaan persamaan matematika . Keamanan informasi setelah dilakukan proses pengiriman dan penerimaan informasi maka dapat dilakukan tindakan – tindakan berikut ini :

1. Membuktikan keaslian adalah proses yang memungkinkan penerima informasi untuk mengetahui asal atau pengirim informasi yang sebenarnya.
2. Menjaga integritas data yaitu proses yang menjamin penerima informasi dapat memeriksa, apakah infomasi telah berubah sebelum diterima.
3. Pembuktian seseorang telah mengirim pesan adalah proses untuk menjamin pengirim informasi tidak dapat menyangkal bahwa dia telah mengirim informasi tersebut.
4. Menjaga kerahasian yaitu proses untuk menjamin informasi yang dikirim tidak dapat dipahami isinya oleh orang yang tidak berhak.

III. METODOLOGI PENELITIAN

Metodologi pengembangan perangkat lunak yang digunakan adalah UP (*Unified Process*) atau dikenal juga dengan proses iteratif dan incremental merupakan sebuah proses pengembangan perangkat lunak yang dilakukan secara iteratif (berulang) dan inkremental (bertahap dengan progres menaik). iteratif bisa dilakukan dalam setiap tahap, atau iteratif tahap pada proses pengembangan perangkat lunak untuk menghasilkan perbaikan fungsi yang incremental (bertambah menaik) di mana setiap iterasi akan memperbaiki iterasi berikutnya. Salah satu Unified Process yang terkenal adalah RUP (*Rational Unified Process*).

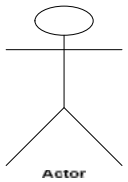
RUP (*Rational Unified Process*) adalah pendekatan pengembangan perangkat lunak yang dilakukan secara berulang-ulang (*iterative*), fokus pada arsitektur (*architectur-centric*), lebih diarahkan berdasarkan penggunaan kasus (*use case driven*) dengan metode pemodelan yang digunakan adalah UML (*Unified Modeling Language*).

UML (*Unified Modeling Language*) adalah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggambarkan diagram dan teks-teks pendukung.

IV. ANALISIS SISTEM

A. Analisis Kebutuhan Sistem

Memiliki kebutuhan sistem yang dapat membantu memudahkan dalam membaca doa harian. Dalam aplikasi ini memiliki 2 aktor yaitu pengirim dan penerima yang dalam hal ini merupakan masyarakat umum atau disebut user yaitu aktor yang menggunakan aplikasi untuk mengirim pesan singkat yang sudah terenkripsi, dan mendeskripsi pesan masuk yang terenkripsi.



Gambar 1. Aktor Sistem

Tabel 1. Deskripsi Kebutuhan Aktor

No.	Aktor	Deskripsi Kebutuhan
1.	Pengirim	1. Tulis pesan 2. Enkripsi
2.	Penerima	1. Terima pesan 2. Deskripsi

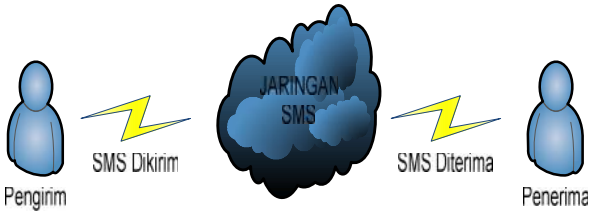
B. Arsitrektur Global Perangkat Lunak

Perangkat lunak yang akan dibangun, merupakan perangkat lunak yang diterapkan pada telepon seluler yang memiliki fungsi untuk melakukan enkripsi SMS. Perangkat Lunak yang akan dibangun harus dapat melakukan pengiriman dan penerimaan pesan.

Perangkat Lunak yang akan dibentuk merupakan perangkat lunak yang akan digunakan untuk berkomunikasi. Oleh karena itu, perangkat lunak yang akan dibangun akan ditanamkan pada pengirim dan juga penerima.

Pengguna akan berinteraksi dengan perangkat lunak melalui user interface yang disediakan oleh perangkat lunak, pengguna memasukan data dengan menggunakan keypad yang dimiliki oleh telepon seluler. Pesan yang telah dibuat dikirimkan ke telepon seluler lain melalui jaringan SMS.

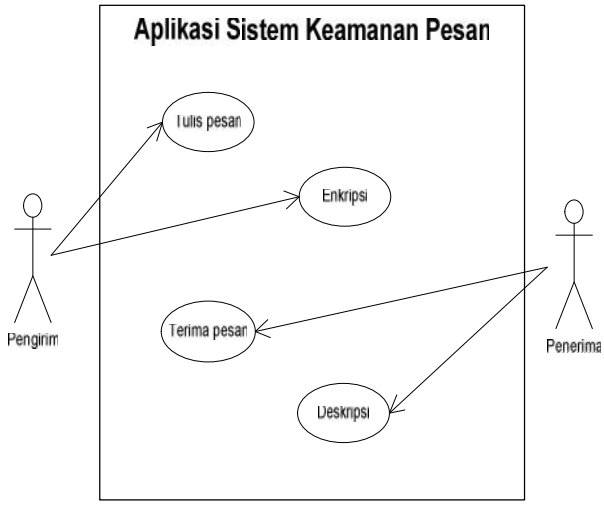
Berikut merupakan arsitektur global aplikasi.



Gambar 2. Arsitektur Global Perangkat Lunak

C. *Use Case Diagram*

Diagram Use Case merupakan bagian tertinggi dari fungsionalitas yang dimiliki sistem yang akan menggambarkan bagaimana seseorang atau aktor akan menggunakan dan memanfaatkan sistem. Diagram ini juga mendeskripsikan apa yang akan dilakukan oleh sistem. Use Case terdiri dari tiga bagian yaitu identifikasi aktor, identifikasi Use Case dan scenario Use Case.



Gambar 3. Use Case Diagram

D. *Activity Diagram*

Activity Diagram merupakan bagian dari penggambaran sistem secara fungsional menjelaskan proses-proses logika atau fungsi yang terimplementasi oleh kode program. Activity Diagram memodelkan event-event yang terjadi didalam suatu Use Case dan digunakan untuk pemodelan aspek dinamis dari system. Aktivitas menggambarkan proses yang berjalan, sementara use case menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

V. IMPLEMENTASI SISTEM

Implementasi sistem berisi tentang dokumentasi sistem yang meliputi spesifikasi minimum kebutuhan untuk implementasi sistem, tampilan layar program dan hasil. Tahap yang harus dilalui setelah melewati tahap perancangan dan pengkodean adalah tahap pengujian. Pengujian terhadap program ini dilakukan dengan tujuan untuk mengetahui apakah program atau sistem yang dibuat berjalan dan berfungsi sesuai dengan rancangan.

A. Tampilan Menu Android

Menu android adalah tampilan muka apabila aplikasi dibuka pertama kali, adapun menu android yang dirancang adalah sebagai berikut:

Tampilan Menu Android

B. Tampilan Menu Aplikasi Sistem Keamanan Pesan

Pada tampilan ini terdapat tiga tombol yang memiliki fungsi masing-masing yaitu: tombol tulis pesan berfungsi untuk menuju activity tulis pesan, tombol kotak masuk berfungsi menuju activity kotak masuk, tombol about berfungsi menuju activity about.

Tampilan Menu Aplikasi Sistem Keamanan Pesan

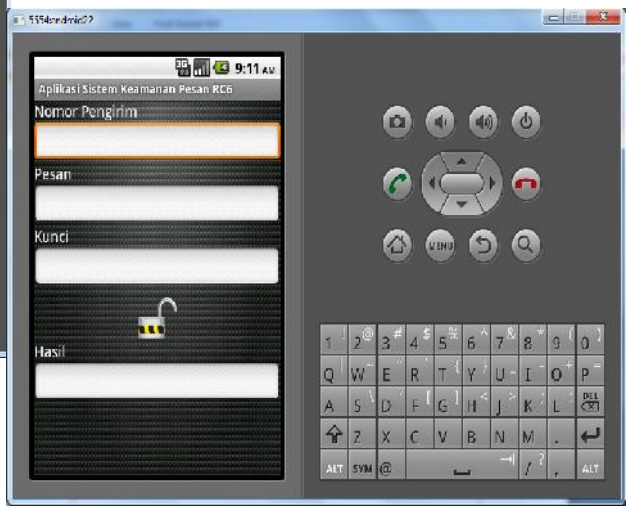
C. Tampilan Tulis Pesan

Pada tab tulis pesan ini pengguna diharapkan untuk memasukan nomor tujuan, pesan yang ingin disampaikan, memasukan kunci pesan, menekan tombol enkripsi kemudian bias akan muncul hasil enkripsi kemudian setelah muncul hasil enkripsi pengguna dapat menekan tombol kirim.

Tampilan Tulis Pesan

D. Tampilan Terima Pesan

Untuk tampilan terima pesan berfungsi membuka pesan yang diterima dalam bentuk ciphertext, lalu didekripsikan menggunakan kunci yang sama. Dimana menu terima pesan terdapat nomor pengirim, isi SMS terenkripsi, lalu kunci, dan hasil seperti ditunjukan pada gambar:



Tampilan Terima Pesan

E. Tampilan About

About merupakan kelas yang berfungsi untuk memberikan informasi tentang aplikasi pengguna, About dapat dilihat dengan cara menekan tombol about pada tampilan menu aplikasi keamanan pesan tampilan aboutnya adalah :

Tampilan About

VII. PENUTUP

A. Kesimpulan

Dari uraian yang terdapat pada laporan ini, maka penulis menarik beberapa kesimpulan sebagai berikut:

1. Dengan menggunakan algoritma RC-6 maka data akan lebih aman karena diterapkan proses enkripsi sehingga tidak dapat dibaca sama orang lain..
2. Dengan aplikasi sistem keamanan pesan setiap orang dapat mengamankan pesan informasinya yang bersifat rahasia baik yang dikirim maupun yang diterima dalam bentuk sms.
3. Dengan aplikasi yang dapat diterapakan pada smartphone berbasis android maka setiap orang akan lebih mudah menggunakan aplikasi ini.

B. Saran

Saran-saran yang dapat penulis berikan terhadap jalannya aplikasi *secure message* adalah:

1. Perlunya sarana penunjang sistem yang berbasis android, baik itu perangkat keras seperti *smartphone* maupun perangkat lunak yang mendukung untuk menjalankan aplikasi ini.
2. Agar aplikasi dapat dipahami oleh pengguna (*user*) yang dalam hal ini adalah masyarakat umum, maka perlu dibuat tampilan panduan atau tutorial agar aplikasi tersebut dapat digunakan atau dioperasikan dengan baik.
3. Pesan akan cenderung membesar, ada kemungkinan sebuah pesan menjadi dua buah pesan setelah dienkripsi.

DAFTAR PUSTAKA

A.S., Rosa, M. Shalahuddin. 2011. *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung : MODULA.

A.S., Rosa, M. Shalahuddin. 2011. *Modul Pembelajaran Pemrograman Berorientasi Objek)*. Bandung : MODULA.

B, Al - Bahra bin Ladjamuddin. 2004. *Konsep Sistem Basis Data dan Implementasinya*. Yogyakarta : GRAHA ILMU.

Defni, Indri Rahmayun. 2014. Enkripsi SMS (Short Message Service) pada Telepon Selular Berbasis Android dengan Metode RC6. *Jurnal Momentum*. Vol. 16 No.1.

Dharwiyanti, Sri dan Romi Satria Wahono. 2003. *Pengantar Unified Modeling Language(UML)*. IlmuKomputer.Com

Maria Polina, S.Kom., M.Sc., Agnes, Drs. Jong Jek Siang, M.Sc. 2005 *Kiat Jitu Menyusun Skripsi Jurusan Informatika / Komputer*. Yogyakarta : ANDI OFFSET.

Marlinda S.Kom, Linda. *Sistem basis data*. 2004. Yogyakarta : ANDI.

Nugroho, Adi. 2005. *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*. Bandung : Informatika Bandung.

Nugroho, Adi. 2007. *Pemrograman Java untuk Aplikasi Basis Data dengan Teknik XP Menggunakan IDE Eclipse*. Yogyakarta : ANDI.

Pudjo Widodo, Prabowo, Herlawati. 2011. *Menggunakan UML*. Bandung : INFORMATIKA.

Safaat H., Nazruddin. 2012. *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Andorid*. Bandung : INFORMATIKA.

Safaat H., Nazruddin. 2013. *Aplikasi Berbasis Andorid*. Bandung : INFORMATIKA.

Sommerville, Ian. 2003. *Software Engeneering (Rekayasa Perangkat Lunak) jilid 1*. Jakarta : Erlangga.

Sommerville, Ian. 2003. *Software Engeneering (Rekayasa Perangkat Lunak) jilid 2*. Jakarta : Erlangga.

Sutabri, Tata. 2012. *Analisis Sistem Informasi*. Yogyakarta : ANDI.

Sutabri, Tata. 2012. *Konsep Sistem Informasi*. Yogyakarta : ANDI.

UU No 11 Tahun 2008. www.kemenag.go.id/file/dokumen/UU1108.pdf.

Yudi Prayudi, IdhamHalik. 2015. Studi dan Analisis Algoritma Rivest Code 6 (RC6) dalam Enkripsi/Dekripsi Data. *SNATI 2005*. ISBN: 979-756-061-6.

Yuyun Priatna. 2006. Perancangan dan Implementasi Algoritma Kriptografi RC6 guna Mengamankan Data Pesan Singkat pada Ponsel yang berbasis J2ME. *Tugas Akhir*. Fakultas Teknik dan Ilmu Komputer. UNIKOM. Bandung.